Учреждение образования Федерации профсоюзов Беларуси

«Международный университет «МИТСО»

Рег. № \_\_\_\_\_\_\_\_\_\_ Кафедра\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_20\_\_\_\_

**КУРСОВАЯ РАБОТА**

на тему \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

по дисциплине \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Основные замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отметка о допуске курсовой работы к защите:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_  Подпись научного руководителя: \_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Студент: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О. – полностью)  Курс \_\_\_\_\_, группа \_\_\_\_\_\_\_\_\_\_\_  Факультет \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Специальность \_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Научный руководитель:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (должность, ученая степень, ученое звание)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О. – полностью) |

Минск 20\_\_\_

Содержание

Введение……………………………………………………………………

Глава 1. Алгоритм разработки……………………………………………

Глава 2. Структура базы данных……………………………………………

Глава 3. Алгоритм работы функции……………………………………….

Глава 4. Инструкция пользователя ……………………………………….

Глава 5. Отладка программного кода и тестирование…………………….

Заключение…………………………………………………………………

Ссылки на источники………………………………………………………

# Введение

На данный момент времени компьютеры прочно вошли в нашу жизнь. На данный момент времени сложно найти человека, у которого не было бы компьютера. При помощи компьютера делаются многие дела и сложно представить почти любую работу чтобы у работника не было компьютера. Отсюда можно сделать весьма легкий вывод, что актуальность сфер деятельности связанных с компьютером еще очень долгое время не исчезнет.

В данной курсовой работе я ставила для себя целью создания проекта на новом для меня языке программирования.  
  
Цель курсовой работы - закрепление теоретического материала и выработка навыков нисходящего проектирования программного модуля учёта читателей в библиотеке, имена, фамилии, книги, которые были взяты.

Текст самой работы:

Разработать программный модуль учета выдачи книг читателям в библиотеке. Программный модуль учитывает кому выдана книга, название книги, срок возврата.  
  
Для достижения поставленной цели необходимо решить следующие задачи:   
  
Разработать алгоритм и структуру данных для решения разработки программный модуль учета книг в библиотеке.  
  
Разработать удобный пользователю интерфейс работы с программной системой (подсказки, сообщения об ошибках и т.д.).  
  
Выполнить кодирование и отладку алгоритма с использованием языка Python c использованием технологий объектно-ориентированного программирования.  
  
Разработать руководство пользователя для работы с программой.  
  
Работа состоит из 5 глав, введения и заключения. В первой главе я рассказала о разработке алгоритма, поиске информации и анализе данных. Во второй главе я описала структуру данных. В третьей главе Алгоритм работы функций. В четвертой главе указано описание интерфейса программы. В пятой главе рассказано об отладке и тестировании программы

Глава 1. Алгоритм разработки

Анализ разработки данных сформировал требования к программе, которые должны быть выполнены в соответствие с требования пользователя.

Постановка задач перед программным модулем:

- Создание процесса авторизации в качестве нового пользователя и по уже существующему пользователю.

- Создание функции просмотра читателей, и книг взятых этими читателями в библиотеке

- Возможность добавление либо удаление читателей.

- Удобство использования.

- Производительность и скорость работы.

- Наличие вспомогательных библиотек.

-Удобный интерфейс для восприятия.

Для выполнения этой задачи был выбран язык программирования Python. И среда разработки Pycharm.

Последующей задачей стала разработка алгоритма.

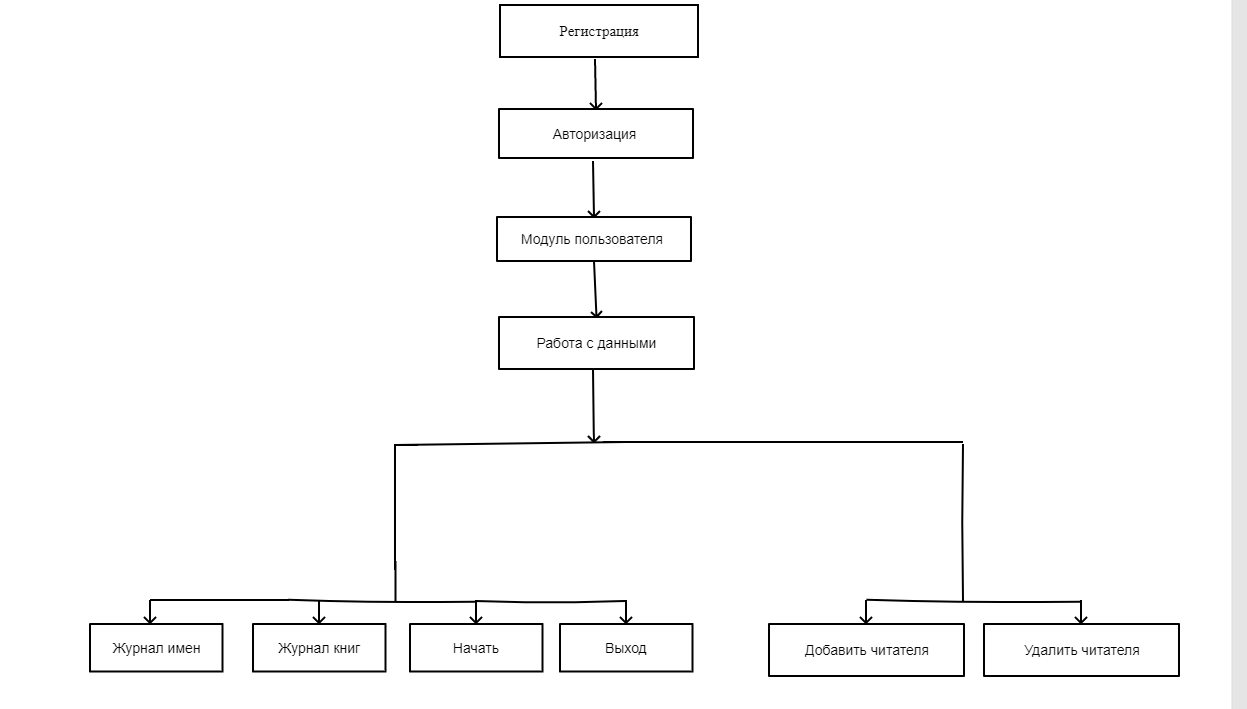
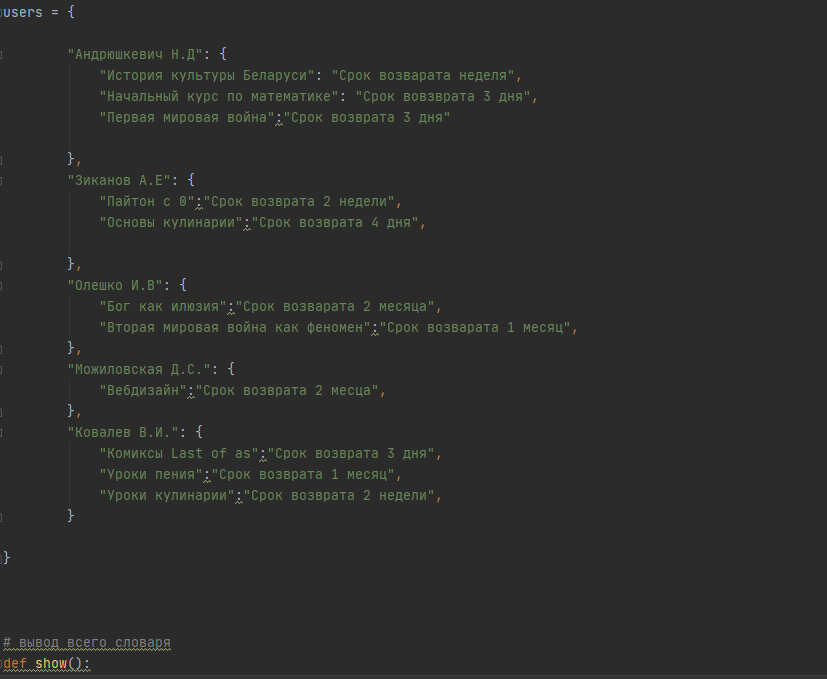


Рисунок 1. Модульная структура программы

Глава 2. Структура базы данных

Структура данных представлена в виде непосредственно словаря, который составляется непосредственно разработчиком, а после может редактироваться пользователем.

В словарь записывается ключ, в данном случае это имена читателей, а книги это строки записанные в этот ключ.



Основными данными являются:

-ФИО

-Книги

-Срок возврата

Глава 3. Алгоритм работы функций

После рассмотренных и выполненных задач, можно приступать к записи алгоритма на языке программирования. Саму программу рассмотрим в виде последовательности:

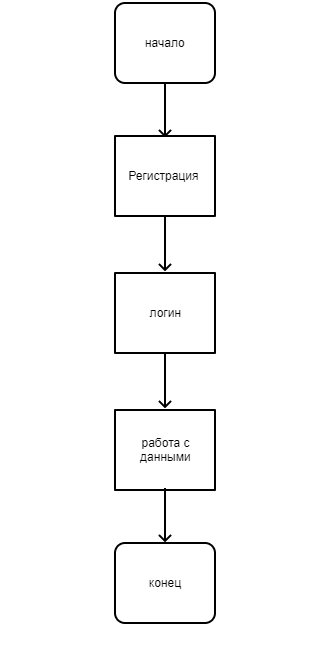


Рисунок 2. Алгоритм функции main

Функциями def registration и def login отвечающие за реализацию кнопок регистрации и логина соответственно, при запуске программы эти методы будут являться первичными.

Рассмотрим функцию def login

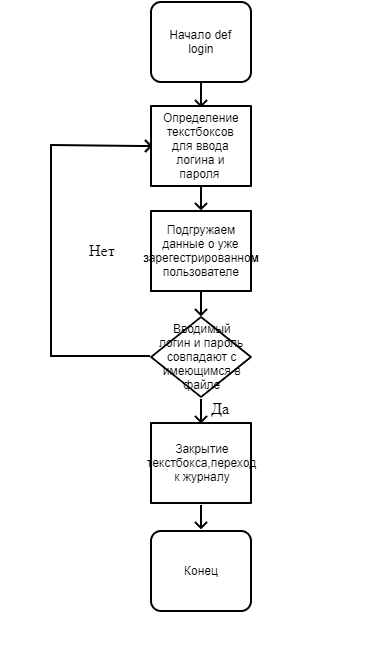


Рисунок 3. Алгоритм метода def login

Функция def udalit отвечает за реализацию кнопки удаление, при вводе имя читателя, программа удаляет этого читателя из словаря.

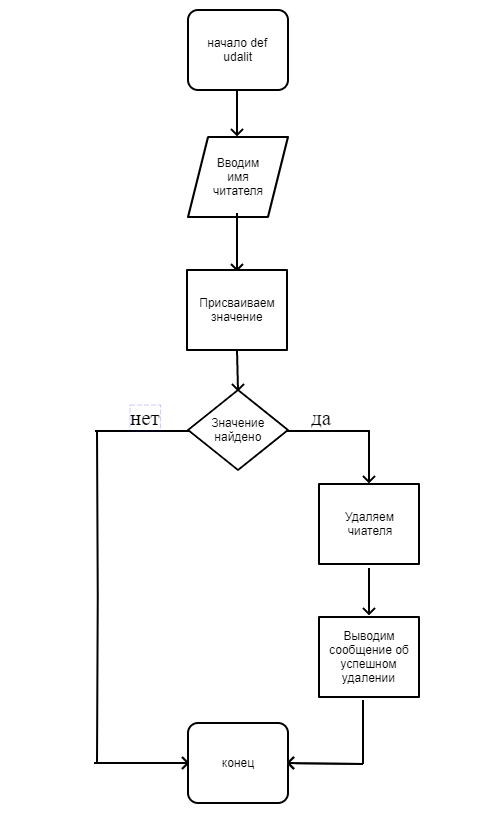


Рисунок 3. Алгоритм метода def udalit

Глава 4. Инструкция пользователя

Последний этап курсовой, создание записки по пользованию программы.

Конечный результат кода можно посмотреть в приложении.

Заметки по использованию:

* Внимательно читайте инструкцию использования.
* Сохраните инструкцию для дальнейшего использования.
* Хранить инструкцию в виде электронного документа и/или в печатном варианте.
* Не удалять и/или уничтожать инструкцию по применению.
* В случае непредвиденных ситуаций полагаться на инструкцию для пользователей.

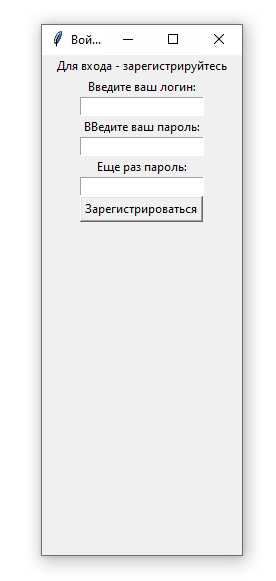
Начало работы

Для начала работы с программой запустите файл под названием «Список читателей библиотеки.pу» щелкнув дважды левой кнопкой мыши по файлу, или один раз правой кнопкой мыши и выбрав в меню функцию открыть. Произойдет запуск программы. Переключитесь на английскую раскладку клавиатуры.

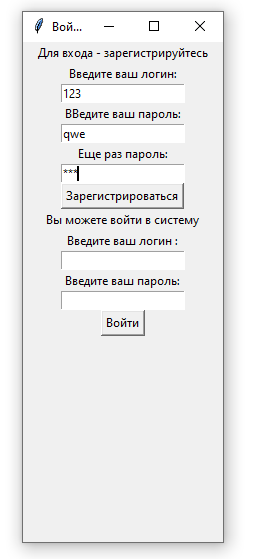
17

Взаимодействие с программой

Для того чтобы запустить программу, нажмите сочетание клавиш «Ctrl+Shift+F10», или правой кнопкой мыши по коду и на название самой программы. При успешном старте программы на экране появиться окно регистрации.

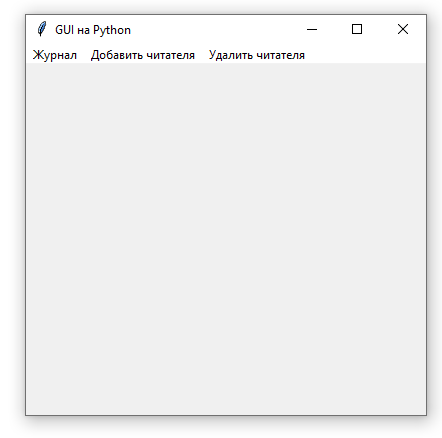


В этом окне требуется зарегистрироваться, ввести логин, пароль 2 раза.

При успешном вводе появится дополнительное окно, окно авторизации. 

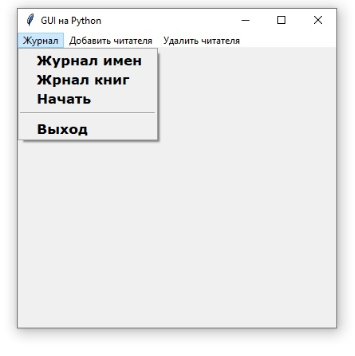
При успешной регистрации появится сообщение об успешном вводе, далее можно закрыть окно регистрации и переходить к следующему окну, окну работы самой программы.

.

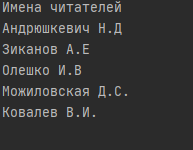


В данном окне мы видим, несколько кнопок, каждая отвечает за свои функции. Разберем каждое по порядку.

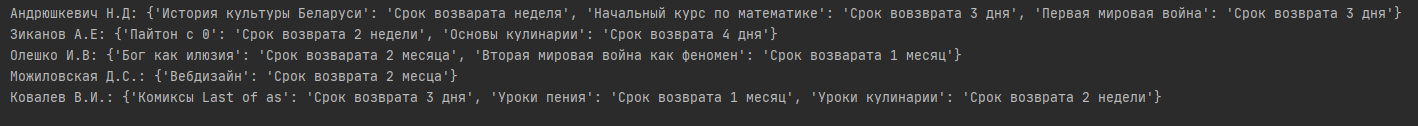
Журнал, кнопка, при ее нажатии высветиться небольшое окно, с еще парой кнопок, действие которых мы также разберем.



Первая кнопка, журнал имен, при ее нажатии нам высветиться все имена читателе, которые зарегистрированы в библиотеке.



Вторая кнопка это-кнопка отвечающая за те книги что взяли читатели при приходе в библиотеку, так же показывается и срок возврата.

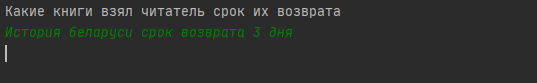


Третья кнопка-это начало самой программы, при ее нажатии мы переходим в консоль, вводим имя читателя которого мы хотим найти.

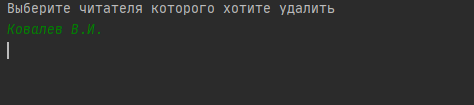


Четвертая кнопка-это завершении программы. При ее нажатии программа выключается.

Пятая кнопка-это кнопка добавление читателя в библиотеку. Сначала просят ввести имя читателя, потом кину которую он взял и срок возврата. После появится сообщение что читатель успешно добавлен.



И последняя кнопка, кнопка удаления читателя. При ее нажатии вас попросят ввести читателя, которого надо удалить из журнала. После ввода и успешного удаление на экране появится сообщение что читатель был успешно удален.



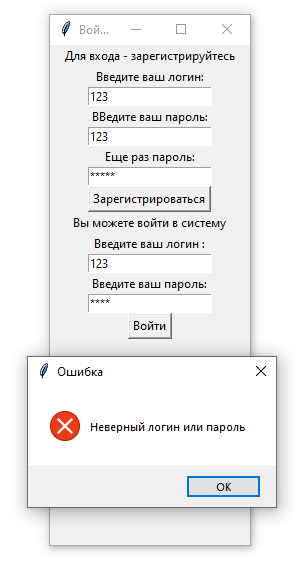
Глава 5. Отладка программного кода и тестирование

Следующим этапом разработки алгоритма стала необходимость в отладке программного кода. Это необходимо делать из-за того что в процессе работы, возникают всякого рода ошибки. Возможности отладчика:

* выполнение пошагового исполнения программы с остановкой после каждого оператора;
* возможность просмотра текущего значения любой переменной или нахождения значения любого выражения;
* в случае необходимости можно задать новое значение переменной;
* можно установить в программе контрольные точки – точки, в которых программа временно прекращает свое выполнение.

Проведя все эти действия можно понять где и в каком месте была ошибка. Вот пару примеров исправленных исключительных ситуаций:

При вводе не существующего пользователя мы получим сообщение об ошибке.



При вводе не существующего пользователя мы получим сообщение в консоли что такого читателя нет в списке.



# Заключение

В заключение хотела бы сказать, что все цели поставленные в курсовой работе были выполнены :  
  
Основная цель была - закрепить теоретический материал и выработать навыки нисходящего проектирования программного модуля читателей в библиотеке, имена, фамилии, книги, которые были взяты.  
  
Все задачи, которые я поставила в начале моей работы были выполнены:

1. Разработать алгоритм и структуру данных для решения разработки программный модуль учета успеваемости студентов.
2. Оценить эффективность алгоритма в условных единицах.
3. Разработать удобный пользователю интерфейс работы с программной системой (подсказки, сообщения об ошибках и т.д.)
4. Выполнить кодирование и отладку алгоритма с использованием языка Python c использованием технологий объектно-ориентированного программирования.
5. Разработать руководство пользователя для работы с программой.

В ходе выполнения курсового проектирования все задачи для достижения цели были успешно выполнены. По окончанию курсового проекта был создан конечный программный продукт, который готов для использования. Качество программного продукта полностью отражает степень усвоения курса дисциплины «Основы конструирования программы» и показывает уровень усвоенных знаний и практических умений. Программа полностью выполняет поставленную задачу.

# Список использованных источников

1. Н. Вирт. Алгоритмы + структуры данных = программы, 1-е издание ДМК Пресс, 1976. – 276 с.
2. С. Рой. Алгоритмы. Теория и практическое применение 2016. – 544 с.
3. А. В. Ахо. Структуры данных и алгоритмы, 1-е издание Издательский дом "Вильямс", 2000. – 384 с.
4. Этапы разработки программы - <https://info-comp.ru/programmirovanie/724-stages-of-program-development.html>
5. Разработка ПО - <https://info-comp.ru/programmirovanie/724-stages-of-program-development.html>
6. Процесс разработки ПО - <https://ru.wikipedia.org/wiki/Процесс_разработки_программного_обеспечения>
7. Сравнение языков программирования - <https://www.cyberforum.ru/cpp-beginners/thread942827.html>
8. <https://ru.wikipedia.org/wiki/Сравнение_языков_программирования>
9. Блок схемы - <https://pro-prof.com/archives/1462>
10. <https://ru.wikipedia.org/wiki/Блок-схема>
11. Дерево решений - <https://loginom.ru/blog/decision-tree-p1>
12. Тестирование решений - <https://habr.com/ru/company/otus/blog/443418/>
13. <https://ru.wikipedia.org/wiki/Тестирование_программного_обеспечения>
14. Сборник мануалов устройств и программ компании MSI -https://www.manualowl.com/c/MSI/Motherboard/Manuals

# Приложение:

import pickle

from tkinter import \*

from tkinter import messagebox

import sys

root = Tk()

root.geometry("200x500")

root.title("Войти в систему")

# Регистрация

def registration():

text = Label(text="Для входа - зарегистрируйтесь")

text\_log = Label(text="Введите ваш логин:")

registr\_login = Entry()

text\_password1 = Label(text="ВВедите ваш пароль:")

registr\_password1 = Entry(show='\*')

text\_password2 = Label(text="Еще раз пароль:")

registr\_password2 = Entry(show="\*")

f = Button(text="Зарегистрироваться", command=lambda: save())

text.pack()

text\_log.pack()

registr\_login.pack()

text\_password1.pack()

registr\_password1.pack()

text\_password2.pack()

registr\_password2.pack()

f.pack()

def save():

login\_pass\_save = {}

login\_pass\_save[registr\_login.get()] = registr\_password1.get()

f = open("login.txt", "wb")

pickle.dump(login\_pass\_save, f)

f.close()

login()

def login():

text\_log = Label(text=" Вы можете войти в систему ")

text\_enter\_login = Label(text="Введите ваш логин :")

enter\_login = Entry()

text\_enter\_pass = Label(text="Введите ваш пароль:")

enter\_password = Entry(show="\*")

button\_enter = Button(text="Войти", command=lambda: log\_pass())

text\_log.pack()

text\_enter\_login.pack()

enter\_login.pack()

text\_enter\_pass.pack()

enter\_password.pack()

button\_enter.pack()

def log\_pass():

f = open("login.txt" , "rb")

a = pickle.load(f)

f.close()

if enter\_login.get() in a:

if enter\_password.get() == a[enter\_login.get()]:

messagebox.showinfo("Привет, добро пожаловать!")

else:

messagebox.showerror("Ошибка", "Неверный логин или пароль")

else:

messagebox.showerror("Ошибка","Неверный логин или пароль")

registration()

root.mainloop()

users = {

"Андрюшкевич Н.Д": {

"История культуры Беларуси": "Срок возварата неделя",

"Начальный курс по математике": "Срок вовзврата 3 дня",

"Первая мировая война":"Срок возврата 3 дня"

},

"Зиканов А.Е": {

"Пайтон с 0":"Срок возврата 2 недели",

"Основы кулинарии":"Срок возврата 4 дня",

},

"Олешко И.В": {

"Бог как илюзия":"Срок возварата 2 месяца",

"Вторая мировая война как феномен":"Срок возварата 1 месяц",

},

"Можиловская Д.С.": {

"Вебдизайн":"Срок возврата 2 месца",

},

"Ковалев В.И.": {

"Комиксы Last of as":"Срок возврата 3 дня",

"Уроки пения":"Срок возврата 1 месяц",

"Уроки кулинарии":"Срок возврата 2 недели",

}

}

# вывод всего словаря

def show():

for key, value in users.items():

print("{0}: {1}".format(key, value))

# вывод всех ключей

def show\_keys():

print("Имена читателей")

for key, value in users.items():

print(key)

# получение значений по ключу

def get\_value():

print('Введите имя читателя в билиотеке!')

key = input(str())

print(users.get(key))

# Добавление в ключ

def dobavit():

print("Добавить ученика")

rt = input(str(""))

print('Какие книги взял читатель срок их возврата')

dobav\_value=input(str())

users.setdefault(rt,dobav\_value)

messagebox.showinfo("Успешно добавили")

# Удалени из клоюча

def udalit():

print("Выберите читателя которого хотите удалить")

udal\_stud = input(str())

del users[udal\_stud]

messagebox.showinfo("Читатель успешно удален!")

#выход из программы

def exit():

sys.exit("выход из программы")

root = Tk()

root.title("GUI на Python")

root.geometry("400x350")

main\_menu = Menu()

file\_menu = Menu(font=("Verdana", 13, "bold"), tearoff=0)

file\_menu.add\_command(label="Журнал имен",command=show\_keys)

file\_menu.add\_command(label="Жрнал книг",command=show)

file\_menu.add\_command(label="Начать",command=get\_value)

file\_menu.add\_separator()

file\_menu.add\_command(label="Выход",command=exit)

main\_menu.add\_cascade(label="Журнал", menu=file\_menu)

main\_menu.add\_cascade(label="Добавить читателя", command=dobavit)

main\_menu.add\_cascade(label="Удалить читателя",command=udalit)

root.config(menu=main\_menu)

root.mainloop()